

AMENDMENTS TO THE SPECIFICATION:

Please amend the paragraph beginning at page 1, line 5, as follows:

The ~~present invention~~ technical disclosure relates to digital pre-distortion in power amplifiers with memory effects.

Please amend the paragraphs beginning at page 2, lines 16-18, as follows:

~~This object is achieved in accordance with the attached claims.~~

Briefly, one or more aspects the present invention ~~is~~ are based on a FIR filter structure including individual look-up tables for the filter taps, where each look-up table represents a discretized memory polynomial. Assuming two look-up tables, the training method ~~in accordance with the invention~~ is based on the observation that the look-up table that compensates for memory effects typically has much smaller elements than the look-up table that compensates for non-linearities. This makes it possible to easily determine a first approximation of this dominating look-up table by simply neglecting the other table. This approximation can in turn be used to determine a first approximation of the table compensating for the memory. This procedure can then be iterated by using the latest approximation of one table for determining a refined approximation of the other table until convergence is reached. An advantage of this method is that in each step only a single table has to be

determined (the other table is assumed to be constant), which is a much simpler process than to determine both tables simultaneously in a single step.

Please amend the paragraph beginning at page 3, line 4, as follows:

The aspect(s) of the invention, together with further objects and advantages thereof, may best be understood by making reference to the following description taken together with the accompanying drawings, in which:

Please amend the paragraphs beginning at page 3, lines 17-19, as follows:

FIG. 6 is a diagram illustrating discretization of polynomials in accordance with an embodiment of the present invention;

FIG. 7 is a block diagram of an ~~exemplary~~example embodiment of a pre-distorter suitable to be trained in accordance with the present invention;

Please amend the paragraph beginning at page 9, line 24, as follows:

FIG. 9 is a flow chart illustrating an ~~exemplary~~example embodiment of the training method in accordance with the present invention.

Please amend the paragraphs beginning at page 6 line 20 through page 7, line 19, as follows:

In the ~~exemplary example~~ embodiment of the ~~present invention~~ illustrated in FIG. 7, the complex input signal $x(n)$ is forwarded to an absolute value block 10 and to a multiplier 12. The absolute value signal from block 10 is forwarded to a look-up table LUT0 representing a sampled version of polynomial T_0 . The corresponding (generally complex) value from look-up table LUT0 is forwarded to multiplier 12, where it multiplies the input signal sample $x(n)$. Input signal $x(n)$ is also forwarded to a delay block D, where it is delayed one or several sample periods for forming a delayed sample $x(n-1)$. This delayed sample is processed in the same way as the non-delayed sample by an absolute value block 10, a multiplier 12 and a look-up table LUT1. However, look-up table LUT1 now represents a sampled version of polynomial T_1 instead of T_0 . As illustrated in FIG. 7, further delays and look-up tables may be included. Finally, the obtained products are added to each other in adders 14 to form the pre-distorted signal $PD(n)$. Look-up tables ~~used in accordance with the present invention~~ make computation in real time much more efficient than the polynomial computation for each sample of the input signal used in [1]. The look-up tables may be updated (by using the training method described below) to keep track of slow changes in the characteristics of the power amplifier.

FIG. 8 is a block diagram of an ~~exemplary example~~ embodiment of a base station including a power amplifier provided with a pre-distorter ~~in accordance~~

~~with the present invention.~~ In FIG. 8 elements that are not necessary for understanding ~~the invention~~ have been omitted. The baseband complex signal $x(n)$ is forwarded to a pre-distorter 30 in accordance with the present invention. The pre-distorted signal $y(n)$ is up-converted to intermediate frequency (IF) in a digital up-converter 32 and converted into an analog signal in a D/A converter 34, which in turn is up-converted to radio frequency (RF) by an analog up-converter 36. The RF signal is forwarded to a power amplifier 38, and the amplified signal is forwarded to an antenna. The amplified RF signal is also forwarded to a feedback down-conversion chain including an analog down-converter 40, an A/D converter 42 and a digital down-converter 44. The down-converted feedback signal $z(n)$ is forwarded to a trainer 46, which also receives the pre-distorted input signal $y(n)$ for determining the look-up tables in pre-distorter 30 in accordance with the mathematical principles described below.

Please amend the paragraph beginning at page 9, line 7, as follows:

~~The present invention suggests~~Rather, a simpler method is suggested for determining look-up tables from equation (4). This method is based on the observation that table T_1 , which takes care of memory effects, typically has elements that are at least an order of magnitude smaller than the elements of table T_0 . Thus, as a first approximation equation (4) may be written as:

$$z(n)T_0(|z(n)|)=y(n)$$

(5)

Please amend the paragraph beginning at page 10, line 15, as follows:

In a simple form ~~of the invention~~, the approximations $T_0^{(1)}$ and $T_1^{(1)}$ may be used directly as the output of the training procedure. However, further refinements may be achieved by repeating the "trick" in equation (7) one or several times. Thus, to obtain a second approximation $T_0^{(2)}$ of T_0 , the first approximation $T_1^{(1)}$ of T_1 is inserted into equation (4). Once again an equation having the same form as equation (5) is obtained. This equation may be solved approximately using, for example, the averaging method described above. A second approximation $T_1^{(2)}$ of T_1 may be obtained in a similar way. This process may be repeated to obtain approximations of higher orders until convergence is reached (i.e. until there are only insignificant changes to the approximations from one iteration to the next). Depending on the required accuracy, convergence is usually obtained after 3-5 iterations. However, if the accuracy requirements are relaxed, acceptable tables may even be obtained after the first or second iteration.

Please amend the paragraph beginning at page 11, line 20, as follows:

FIG. 9 is a flow chart illustrating an ~~exemplary example~~ embodiment of the training method ~~in accordance with the present invention~~. Step S1 sets the elements of table T_1 to predetermined values, typically 0 for a first time training and the current T_1 values for an update. Then T_0 is estimated by solving equation (6). Step S2 sets T_0 the determined estimate in equation (4) for

obtaining an equation having the same form as equation (5), This equation is solved for an estimate of T_1 . If simplicity is more important than accuracy, it is possible to stop here and output the obtained estimates as the final tables. However, preferably a few more iterations (steps S3-S5) are performed. Step S3 sets T_1 to the latest determined estimate of T_1 . Then the estimate of T_0 is refined by solving an equation similar to equation (5). Step S4 sets T_0 the latest determined estimate of T_0 . Then the estimate of T_1 is refined by solving an equation similar to equation (5). (Another simple embodiment may be obtained by stopping after step S3. This embodiment involves two iterations for T_0 , but only one iteration for T_1 . This embodiment may, for example, suffice for an update, since a rather good estimate is already available for T_1 in this case.) Step S5 tests whether the tables have converged. This can be done, for example, by summing the absolute values (or squares) of the differences between corresponding elements of the current estimates and the previous estimates and testing whether the obtained sum is smaller than a predetermined threshold. If the tables have converged, the current estimates are provided as the final tables in step S6. If the tables have not converged, steps S3-S5 are repeated. Typically this method is implemented by a micro processor or a micro/signal processor combination and corresponding software.

Please amend the paragraph beginning at page 13, line 1, as follows:

The ~~merits of this invention~~ advantages are at least three-fold: Firstly, already implemented single-table algorithms may be used for both the first table and the second memory table (and possibly further tables). Secondly, an iterative method may be used which usually is very attractive to implement in software due to its simple structure. Thirdly, exhausting computations involving inversion of complex-valued matrices and matrix-matrix multiplications as in the method of Least Mean Squares can be avoided entirely. The present disclosure makes it possible to implement multi-table calculations in processors of the same size as used for single-tables. The execution time will only be slightly longer than for a single-table. Usually only a few loops in the iteration scheme will be necessary to obtain the same convergence as the Least-mean Squares algorithm.